Fig. 1 Architecture | 100

Internal E-mail Infrastructure

Out | 102

| 104

In | 106

Pre-processed Queue

Out

Message Translation

Out

Delivery Queue

In

In

R_1 Ok | 108

| 120

| 118

| 110

Ok?

R_x

R_3 Defer

SMTP (w/Security)

| 112

116

Reflexion Security

R_2 Reject

P(s,r)

| 114

In

Delivery of Out

Other Networks
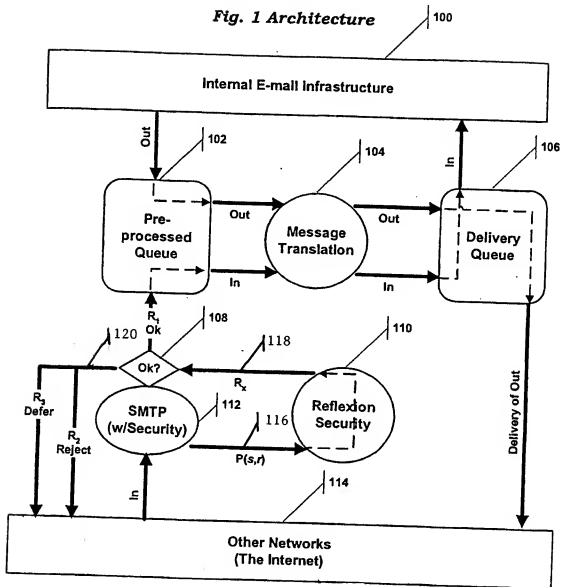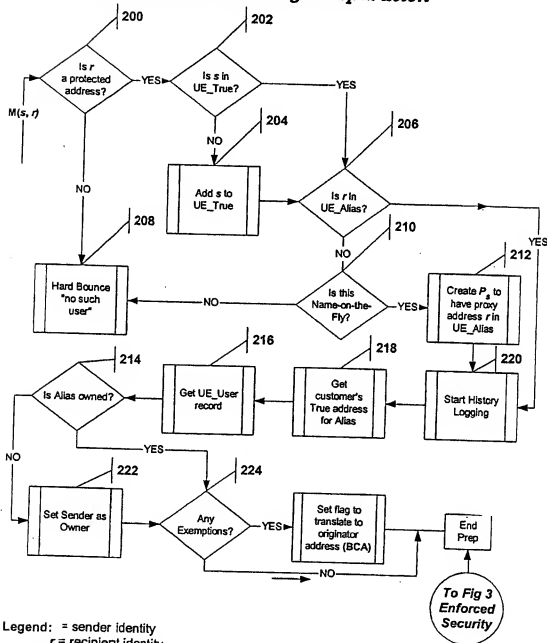(The Internet)

Legend: s = sender identity
r = recipient identity
P(s,r) = Request security status on a message from s to r
R_x = Security status on a message from s to r
R_1 = Ok, continue processing message
R_2 = Reject, do not process the message
R_3 = Defer, temporarily defer the message back to the sending server

Fig. 2 Inbound Message Preparation

Legend: = sender identity
- r = recipient identity
- $M(s,r)$ = A message from s to r
- UE_TRUE is a database table containing "real" (i.e. non-proxy) addresses
- UE_ALIAS is a database table containing proxy addresses
- UE_User is a database table containing user information
- BCA = "Business Card Address", the originator address managed by the internal mail transport agent (i. e. mail server)
- P_s is the security settings for the proxy address registered to s for user that owns originator address to which proxy r is a substitute

Fig. 3 Enforced Security

Flowchart text elements:

- 316 — MUST BE
- 308 — Is P_s "disabled"?
- Denial message to s
- 306 — NO
- 314 — Is P_s "no share"? — YES
- 304 — Is P_s "protected"? — NO
- 324 — Is s exempt? — NO
- 332 — Is s at right domain + DLS on? — NO
- 338 — NO
- To Fig 5 Address Translation
- 332 — Is s authorized to send to r?
- 322 — Send s reminder (to use P_s) — AND
- 302 — Is s exempt? — YES
- 312 — Is P_s "public"? — NO
- 320 — Is P_s "protected"? — NO
- 328 — Is P_s "protected"? — NO
- Is P_s "disabled"? — NO — MUST BE
- Denial message to s
- Is P_s "no share"? — YES
- 300 — Is P_s "Public"? — YES
- Retrieve P_r from P_s from UE_Alias
- Is security enforced? — YES / NO
- M(s, r) — To Fig 4 Flag Security
- Message to s to resend to P_s

Legend:

s = sender identity
r = recipient identity
P_s is the security settings for the proxy address
registered to s for user that owns proxy r
P_r is the security settings for the proxy address r

M(s,r) = A message from s to r
UE_Alias is a database table containing proxy addresses
DLS stands for Domain Level Sharing
Note: it is possible for P_s to be the same object as P_r

Fig. 4

Flag Security

408 — Is P_r "disabled"? — NO

406 — Is P_r "no share"? — NO → MUST BE

404 — Is P_r "protected"? — NO

402 — Is P_s "public" or "protected"? — NO

400 — Is P_s exempt? — YES

Is P_r Public?

Retrieve P_r, P_s from UE_Alias

M(s, r)

414 — Is s exempt? — YES

412 — Is P_s "no share" or "disabled"? — NO

Is P_s "no share" or "disabled"? — MUST BE

Flag subject line

420 — Is s authorized to send to r? — NO

424 — Is s all right domain + DLS on? — NO

Flag subject line

422 — Flag subject line

To Fig 5 Address Translation

YES — YES — YES

M(s,r) = A message from s to r
UE_Alias is a database table containing proxy addresses
DLS stands for Domain Level Sharing
Note: It is possible for P_s to be the same object as P_r

Legend:

s = sender identity
r = recipient identity
P_s is the security settings for the proxy address
   registered to s for user that owns proxy r
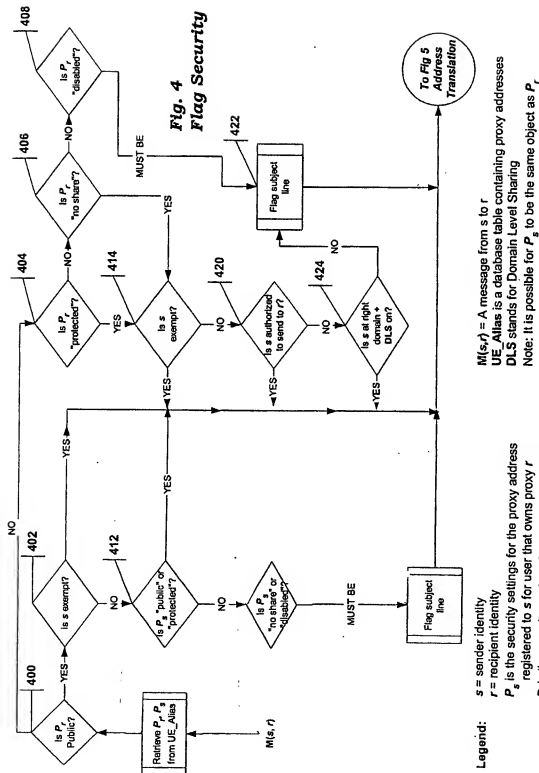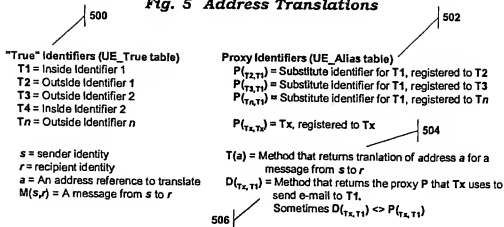P_r is the security settings for the proxy address r

Fig. 5   Address Translations

500   502

"True" Identifiers (UE_True table)
T1 = Inside Identifier 1
T2 = Outside Identifier 1
T3 = Outside Identifier 2
T4 = Inside Identifier 2
Tn = Outside Identifier n

s = sender identity
r = recipient identity
a = An address reference to translate
$M(s,r)$ = A message from s to r

Proxy Identifiers (UE_Alias table)
$P_{(T2,T1)}$ = Substitute identifier for T1, registered to T2
$P_{(T3,T1)}$ = Substitute identifier for T1, registered to T3
$P_{(Tn,T1)}$ = Substitute identifier for T1, registered to Tn

$P_{(Tx,Tn)}$ = Tx, registered to Tx   504

$T(a)$ = Method that returns tranlation of address a for a
message from s to r
$D_{(Tx,T1)}$ = Method that returns the proxy P that Tx uses to
send e-mail to T1.
Sometimes $D_{(Tx,T1)}$ <> $P_{(Tx,T1)}$

506

INBOUND, successfully past security, where:

1. $a = r$, s = T2, $r = P_{(T2,T1)}$, then $T(a)$ = T1

2. $a = r$, s = T2, $r = P_{(T3,T1)}$, then $T(a)$ = T1

3. $a = P_{(T4,T4)}$, s = T2, $r = P_{(T2,T1)}$, then $T(a)$ = T4

4. $a = P_{(T4,T4)}$, s = T2, $r = P_{(T3,T1)}$, then $T(a)$ = T4

5. a = T3, s = T2, $r = P_{(Tx,T1)}$, then $T(a)$ = T3

6. $a = P_{(Tx,Ty)}$, s = T2, T2 is exempt, r = any P, then $T(a)$ = Ty

OUTBOUND, no security on outbound, where:

7. $a = r$, s = T1, r = T2, then $T(a) = P_{(T2,T1)}$

8. $a = r$, s = T1, r = T2, $D_{(T2,T1)}$ <> $P_{(T2,T1)}$, then $T(a) = D_{(T2,T1)}$

9. $a = r$, s = T1, r = T2, $D_{(T2,T1)} = P_{(T2,T1)}$, then $T(a) = P_{(T2,T1)}$

10. $a = r$, s = T1, r = T2, r is exempt, then $T(a) = P_{(T1,T1)} [s]$

11. a = T3, s = T1, r = T2, then $T(a) = P_{(T3,T1)}$

12. a = T3, s = T1, r = T2, $D_{(T3,T1)}$ <> $P_{(T3,T1)}$, then $T(a) = D_{(T3,T1)}$

13. a = T3, s = T1, r = T2, $D_{(T1,T2)} = P_{(T2,T1)}$, then $T(a) = P_{(T3,T1)}$

14. a = T3, s = T1, r = T2, T3 is exempt, then $T(a) = P_{(T1,T1)} [s]$